ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

Realtime Operating System Implementation on AVR XMEGA For Unmanned Aerial Vehicle Autopilot

Roghib Muhammad Hujja, Raden Sumiharto, Setyawan Bekti W.

Lecturer at Electronic and Instrumentation Program, Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia
Lecturer at Electronic and Instrumentation Program, Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia
Lecturer at Mechanical Engineering Program, Vocational Department, Universitas Gadjah Mada, Yogyakarta, Indonesia

ABSTRACT: An RTOS (Realtime Operating System) for Autopilot has been implemented for processing important task such as reading sensor, processing movement data, and continuous data transmission by using XMEGA microcontroller. The data angle roll, pitch and yaw from the IMU (Inertial Measurement Unit) sensor then transmitted to the ground station using XBEE. FreeRTOS used as the basic operating system with three main tasks that is, read sensor, control, and send data. Priority scheme used on the system with the highest are control task then read sensor task and the lowest is send data task. The system has two control modes, manual mode uses PWM input from the RC Receiver and forwarded directly to the servo. Auto mode uses IMU sensor readings and process position automatically to maintain a stable horizontally position of the UAV (Unmanned Aerial Vehicle).

KEYWORDS: RTOS, XMEGA, IMU, Autopilot, PWM, UAV, PID.

I. INTRODUCTION

A UAV (Unmanned Aerial Vehicle) requires a control system that performs an overall control in order to accomplish a mission. The control system is called Autopilot. Autopilot controls the maneuvers so the UAV able to move independently to accomplish a mission. In addition, Autopilot also can use sensor data to keep the UAV stable on the track automatically. Autopilot also is equipped with the procedures for a particular event or intelligent systems for independent learning to new conditions [1].

Autopilot system is integration between two systems, the air navigation system using ground-based radio or satellite for position guidance and an aircraft control system that regulates state and movement of aircraft such as position, altitude, speed, angle, and direction. An autopilot system that appropriately integrated with the aircraft flight control system can achieve a fast and accurate response also can maintain the motion stability in a flight path [1].

Aerial vehicles require a fast response in movement control because a small delay will result in a fatal condition such as stall (the crash due to lack of lift) [2]. The speed of response requirement on autopilot can be met by using an RTOS (real time operating system). RTOS has a function to perform multitasking so that some processing can be done at almost the same time. RTOS can handle complex timing processes for each of the UAV so it can be done quickly by the main processor [3].

II. RELATED WORK

Research involving the use of XMEGA microcontroller has been done by some researchers including RTOS usage, timer controlling, PWM (Pulse Width Modulation), input, output, and clock frequency. In addition, also taken reference research on the manufacture of AutoPilot for control and use of sensors in the UAV that has been done before.

The first research is design and implementation of PID controller for lateral and longitudinal motion stability in UAV have been done before [4]. PID controller is used to controlling the movement of the plane so that it remains stable in

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

the desired position, especially on the lateral and longitudinal motion. In the lateral motion, there are two deflection controls that affect the airplane response of the aileron to adjust the angle of the roll and rudder to adjust the yaw angle, whereas in the longitudinal motion there is only one control that affects the aircraft's elevator response to adjusting the pitch angle. The PID control works on a closed-loop system where feedback in the form of an error is received by the controller. With the PID algorithm, the error is used to obtain a control signal (manipulative variable) to compensate for errors. PID controller can be obtained by calculating the condition of aircraft movement to get the coefficient Kp, Ki, and Kd to generate appropriate feedback for controlled UAV [4].

Then, another research has succeeded in making the RTOS-based AutoPilot system using PIC microcontroller (peripheral interface controller) [3]. The system created includes controlling the UAV through ground segment as well as the use of auto and manual control systems. Communication between UAV and ground segment using serial with XBee intermediaries [3]. The AutoPilot is the main system of aircraft control, has 2 working status of command (CMD) status and control wheel steering (CWS) status [5]. CWS is a condition that requires the pilot to control the aircraft manually, with guidance from AutoPilot, often also called semi-autopilot. While the command state is a condition where the plane is fully controlled by AutoPilot. AutoPilot optimization system can be made simulated on Matlab to see the response system.

Next research has succeeded in making small UAVs. The main part consists of sensors, microprocessors, and switching circuits. The section is integrated on a circuit board, where most small UAVs do not have an actuator for the ground. GCS (Ground Control Station) communicates with AutoPilot via data link prior to take-off to manage flight path, control flight parameters, and can also exchange data such as flight path through the data link. AutoPilot can receive commands from the remote control, by changing the switch from autonomous mode to manual mode. From various AutoPilot variations, it can be found that AutoPilot is GPS based and integrated with inertial navigation to improve the reliability of AutoPilot. Small UAVs require small, light, low-power, and high-intelligence hardware [6].

The research about multitasking finally succeeded in applying RTOS on AVR microcontroller. RTOS is used for some tasks for testing using led, keypad, and LCD. The system used is preempted task by using the semaphore to control the change of task.Then, Arroyo conducted research for the use of ADC (Analog to Digital Converter) and PWM on XMEGA microcontroller using AVR Studio programming tools. In the experiment, the ADC input is enabled on XMEGA by enabling the ADC channel register control in XMEGA. Next Arroyo uses a timer/counter on XMEGA and generates the PWM as desired to control multiple servos. The servo movement is governed by the input of the ADC [7]. Yuriy research successfully developed FreeRTOS framework on XMEGA microcontroller [8]. The framework system created includes porting, registers, and memory allocation (heap). An example that Yuriy has made is the use of RTOS for XMEGA SPI interrupt demonstrating how to use RTOS framework on XMEGA microcontroller for SPI interrupt. Another example is how to use XMEGA USART with an RTOS framework for RFID reader applications [9].

Prima in his research has made a prototype of water data, attitude, and heading (ADAHRS) system for fixed-wing unmanned aircraft [10]. ADAHRS has a role in providing flight parameter data which will be used by autopilot module or another module in UAV. Parameters read by ADAHRS are yaw angle, pitch angle, roll angle, as well as elevation, speed, temperature, pressure, and GPS coordinate data. The data obtained can be used on autopilot to control the UAV [10].

III. SYSTEM DESIGN

The autopilot system implemented in the form of hardware that can control the movement of UAV using IMU data and transmits the data to the ground station using serial communication. The controlled part is aileron, rudder, and elevator. The system using an XMEGA microcontroller associated with several sections as shown in Figure 1.

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

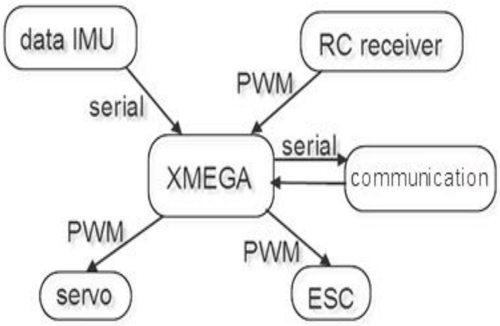

Figure 1. Autopilot system of the relationship block diagram

Relationship block diagram in Figure 1 refers to research on autopilot for small UAVs [4]. Autopilot system has the following main functions:

- Retrieving data from the IMU sensor (x, y, z, height).
- Retrieving data from the RC PWM controller.
- Maintain the stability when flying using mathematical calculations.
- Sending data via serial communication condition of the aircraft.

The Autopilot system made using RTOS need to create some basic task such as task control, sensor reading, and send data. The task run at the almost same time so the process of each task doesn't disturb another task process. There XMEGA is used as the main processor because have many ports that can be used to connect each component in figure 1. Some component such as servo, ESC and RC receiver using PWM as data communication so it can e connected to XMEGA PWM pin. Then IMU and communication (XBEE) using serial can be connected to UART pin on XMEGA.

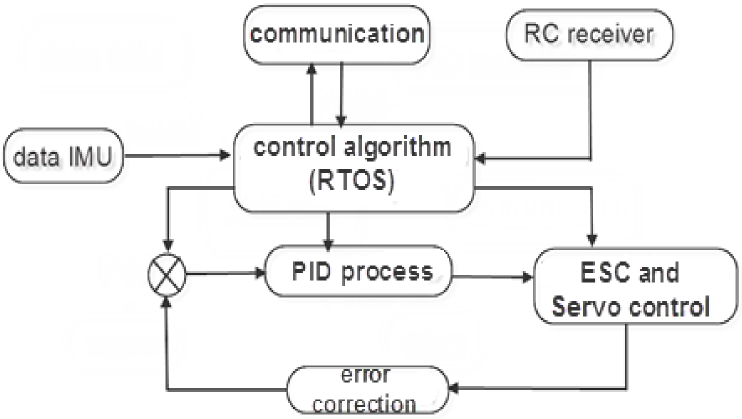

Figure 2. The flow of processing data on Autopilot

Figure 2 is shown Autopilot process flows. In the control algorithm, all data is processed and treated to a pre-programmed mission. Controller read IMU data through the RTOS and the sensor reading task parse and process the data into a global variable so another process can use the data for transmitting and control. Data transmitted to the ground segment via serial communication and the process handled by a separate task that is the send data task. The Data also used as input for controlling UAV movement through PID processing. After processed by the PID controller, data immediately used to control servo and ESC is in the control task. The feedback control obtained will be used to correct an error so the desired condition of the aircraft position can be implemented. In addition, there is PWM input from RC Receiver and used to switch controls between auto and manual. The auto control will be handled by the PID and the position of the aircraft will be maintained stable horizontally. And for manual mode, the control was taken over by the RC Receiver so the UAV can be controlled directly by the RC Transmitter.

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

IV. IMPLEMENTATION

A. XMEGA Minimal System

The main controller the system needs minimum hardware to communicate with input from the sensors and payload also can control the output. The systems are designed for XMEGA can perform communication according to the block diagram shown in Figure 1.

B. Power Source

XMEGA minimal system uses low voltage consumption at 3.3V while for the servo and RC Receiver requires 5V. The power source design that can meet the needs of Autopilot system as shown in Figure 3.

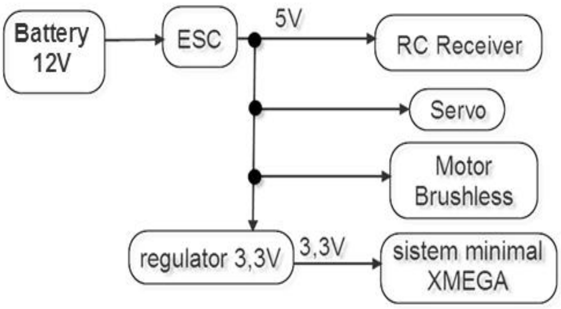

Figure 3. Block diagram design of the system resources

The system uses a 12V battery that connected to the ESC and output of the ESC is 5V. ESC output can be used as power source for RC receiver, servo ,and brushless motors. And also a 3.3V regulator will be used then the power output is safe for minimal system XMEGA. Regulator IC 5V and 12V also used when the voltage too high. Lower voltage can be obtained by connecting the regulator 12V output to regulator 5V input and then connect to regulator 3.3V. with sequential connection, the voltage regulator is not burdened with a too high voltage that can result in overheating. The pin header is used as a connection to other devices that require power such as the servo.

C. FreeRTOS on XMEGA

The main operating system on XMEGA using real time operating system (FreeRTOS) with the preemptive scheme. The systems will be faster in responding the response of the connected devices to the system minimal. By using the preemptive, higher priority task would be an advantage compared with the low priority task [5]. At least three tasks were made for this system is the sensor reading task that serves to make the reading of data IMU via serial communication and IMU data parsing is done so the obtained data can be used by another process. The send task is sending the UAV condition data to the ground station at a certain time interval. And control task is used to perform the primary processing data from both IMU data and data from the RC receiver and directly control the servo and ESC. In Figure 4 can be seen that the priority of the task will be used.

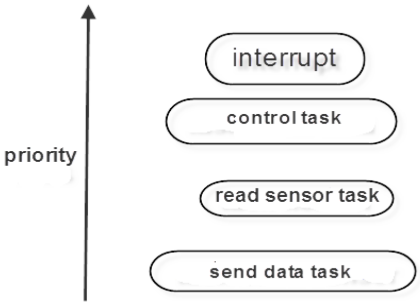

Figure 4. Priority task on Autopilot

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

Priorities and system design for RTOS refer to research that uses FreeRTOS to make Autopilot with XMEGA microcontroller [3] [6]. The main system for RTOS task cycle can be seen in the state diagram in Figure 5.

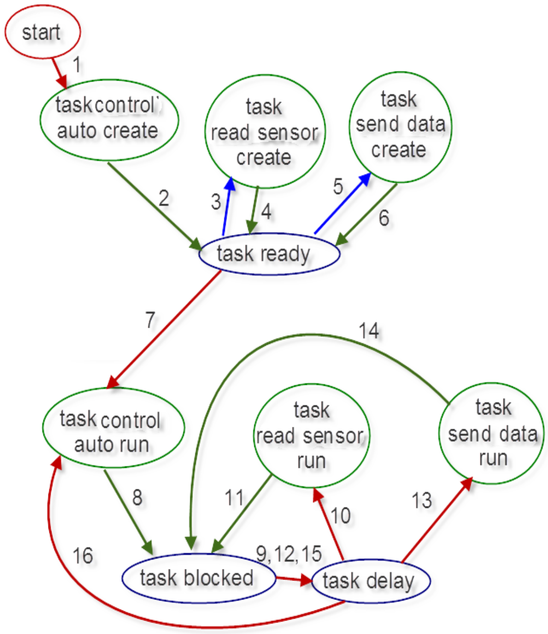

Figure 5. The state diagram of the task on the RTOS

The major systems work cycle as shown in the state sequence diagram in Figure 5. At first task system do task create, that is task control auto. After the task ready then it follows task create for task read sensor and task send data. After all tasks ready then the auto control task executed one cycle until it reaches the delay function and the task will be blocked until the end of delay, continued with the task and the task read sensor, send data run and in accordance with the frequency. Thus each task will perform a continuous loop on the task run into the blocked task and task delay. And with the highest priority task has been completed and delay will be implemented first, although another task is being executed.

V. ANALYSIS

A. Testing PWM Input

Direct PWM data monitoring used as Data testing PWM from RC transmitter used to trigger servo. Maximal and minimal value for each part can be seen in Table 1.

Table 1. Interval PWM data to control servo

Part	Min	Max
Throttle	114	5455
Aileron	94	5457
Elevator	89	5446
Rudder	96	5446

From data on Table 1, PWM minimum and maximum deserve to PWM value that can be used for control servo.

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

B. RTOS test

Task timing testing used as the parameter to RTOS test. Scheduling task running can be seen in Figure 6.

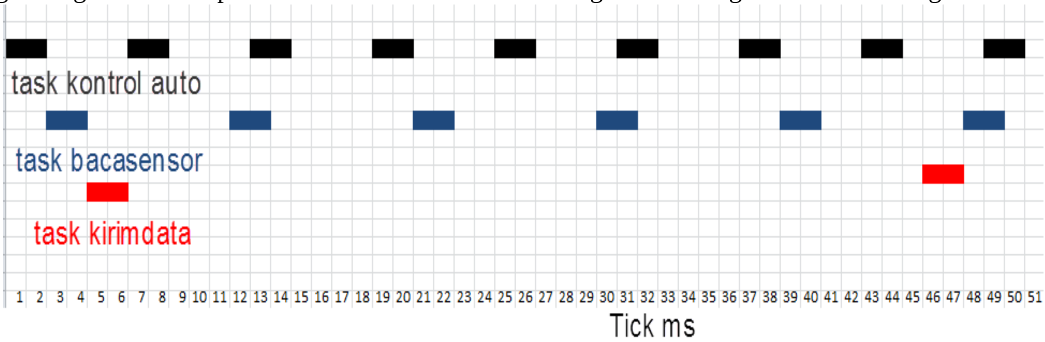

Figure 6. Task running on FreeRTOS

Figure 6 is shown timing tasks that created. Sampled tick shown from tick 1 to tick 51. Task control auto run with configuration delay 4/1000 tick or 250 Hz, task read sensor with delay 8/1000 tick or 125 Hz and task send data with delay configuration 25/1000 or 40Hz. On FreeRTOS used 1000 tick each second and set in FreeRTOSConfig.h. Then delay task sends data measured with a stopwatch to know adequate timing between stopwatch and microcontroller. On Table 2 can be seen result timing measurement of task send data.

Table 2. Send data speed

Data (every10 data)	Send Frequency					
	10 Hz		20 Hz		40 Hz	
	Time(s)	data/s	Time(s)	data/s	Time(s)	data/s
1	19,67	1,967	9,74	0,974	4,84	0,484
2	17,85	1,785	9,82	0,982	4,91	0,491
3	18,23	1,823	9,41	0,941	4,90	0,49
4	18,76	1,876	9,81	0,981	4,20	0,42
5	18,71	1,871	9.89	0,989	4,90	0,49

The Result from Table 2 that is sending data with frequency 10Hz gets average 2 seconds every data transfer. With 20 Hz frequency get average 1 second every data transfer. And with 40 Hz frequency get average 0,5 second every data transfer. From the result then used send data frequency 20 Hz to get task loop data every second. Using increment variable on every task and the value of variable send using task send data that executed every 1 second then loop task data can be measured as shown in Table 3.

Table 3. Loop task data on 1 second

Task	Loop every 1 second	Frekuensi Delay
Control auto	6060	250Hz
Read sensor	7	125Hz
Send data	1	40Hz

The Result from the data seen that task control auto loop 6060 times every 1 second as for task read sensor only 7 loops. This can be the result of task control auto as highest priority. In the next test, the highest priority set to task read sensor and from the result get that task control auto loop decreased for task read sensor loop still have average 7 loops. This can be the result of a pre-emptive system where running time task read sensor and task control auto overlap then the

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 5, Issue 5 , May 2018

highest priority will be executed for lower priority blocked. And also can be seen that task read sensor always stable at average 7 loops each second. From the testing then assumed the optimal configuration for each task in this autopilot system shown in Table 4.

Table 4. Optimal Priority for the task in system

task	Priority	Frequency tick
Control auto	1	250Hz
Read sensor	2	125 Hz
Send data	3	40 Hz

VI. CONCLUSION

Autopilot system using RTOS framework on microcontroller XMEGA implemented and divided into two main parts that are:

- Hardware, that is minimal system XMEGA with input/output pin that supports every device used in the system, a power supply that can give safe power to every device in the system, and a testing RC plane.
- Software, that is FreeRTOS framework with 3 main tasks for reading data sensor, control process, and send data. Also, ground segment interface that used to receive position data of plane.

Optimal task priority from the highest is task control auto, task read sensor, and task sends data.

VII. FUTURE WORK

Suggestion for future development as follows:
- Better power system with security to prevent short circuit.
- Using higher clock frequency with external crystal in result to higher tick rate.
- Better timing delay for every task to prevent overlapping task
- Additional filter in reading sensor to prevent error data reading

REFERENCES

[1] Faidi, M., 2011, System Autopilot Pesawat Terbang, *http://www.gloopic.net/artikel-penerbangan/sarana/sistem-autopilot-pesawat-terbang.html*, 2-5-2011, accessed 10-3-2018.

[2] Nugroho, F., 2008, Prinsip Penerbangan Hukum Bernouli, *http://www.ilmuterbang.com/artikel-mainmenu-29/teori-penerbangan-mainmenu-68/26-private-pilot/86-aerodinamika-hukum-bernoulli,* 11-6-2008, accessed 1-3 2018

[3] Pyke, T., 2009, RTOS PILOT, *http://www.gluonpilot.com/wiki/RTOS_pilot*, 20 -12-2009, diakses accessed 1-3-2018.

[4] Rinaldi, A. A., 2012, Perancangan dan implementasi kontroler PID untuk kestabilan gerak lateral dan longitudinal Unmanned aerial vehicles (UAV) . Jurusan Teknik Elektro, FTI – Institut Teknologi Sepuluh Nopember Kampus ITS, Keputih-Sukolilo, Surabaya.

[5] Shu, J., Gao, J. L., 2010, Design and optimization of autopilot system of flight simulator, international conference on optoelectronic and image processing. Tianjin China.

[6] Tu, H., Du, X., 2010, The design of small UAV autopilot hardware system based on DSP, International conference on intelegent computation technology and automation, Beijing China.

[7] Leksono, B. P., Setiawan, I., Setiyono, B., 2010, Penerapan Real Time Operating Systems (Rtos) Pada Mikrokontroler Avr (Studi Kasus Chibios/Rt). Makalah Seminar Tugas Akhir Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro.

[8] Arroyo, A. A., 2011, Atmel AVR Xmega Code ADC & PWM, http://www.mil.ufl.edu/5666/handouts/AVR_ATXMega128A1_ADC_PWM.pdf, 10-11-2011, accessed 6-3- 2018

[9] Kulikov, Y., 2011, freertos on xmega wiki, *https://docs.google.com/document/pub?id=13GdZZ5kcIWmPArHwJmD9q9AqVGN_CRyPIkId_sJAzVY*, 3 -6-2011, accessed 10-3- 2018.

[10] Prima, A., 2012, Purwarupa sistem referensi air data, attitude, and heading untuk pesawat terbang tanpa awak sayap tetap, Skripsi Elektronika dan Instrumentasi, Fakultas MIPA, Universitas Gadjah Mada.